18.4 Improving CDR Performance via Estimation

Haechang Lee[1], Akash Bansal[2], Yohan Frans[2], Jared Zerbe[2], Stefanos Sidiropoulos[3], Mark Horowitz[1]

[1]Stanford University, Stanford, CA
[2]Rambus, Los Altos, CA
[3]Aeluros, Mountain View, CA

An implementation of the semi-digital dual-loop first-order CDR of [1] is shown in Fig. 18.4.1. The CDR (peripheral) loop consists of a bang-bang phase detector, gain (pre_filt), binary accumulator (P_acc), and phase DAC. Pre_filt is an accumulator that generates a positive or negative carry when the accumulated error reaches a certain programmable threshold (±1, 2, 4, or 8) and acts as a linear attenuator (1, 1/2, 1/4, or 1/8). The truncation occurring in pre_filt also helps to suppress limit cycles caused by loop delay [2]. The core loop is both a frequency synthesizer and multiphase generator. The high-frequency jitter tolerance (i.e. timing margin) of this CDR improves as its bandwidth (loop gain) is reduced since more ISI jitter is removed (Fig. 18.4.5(a)). However, this increases lock time and decreases frequency operation range.

Building an estimator that adjusts its loop gain according to operating conditions can remove this tradeoff. Since the first-order CDR is a phase-domain delta modulator, this estimator is fashioned after the adaptive delta modulator (ADM) that adjusts its loop gain based on the time history of the quantizer output. A modified adaptation is required due to the loop delay (5 cycles) of the CDR. The adaptive block (adapt) accumulates the output of pre_filt. Every N (e.g. 32) cycles, the absolute value of the accumulator (abs_n) is compared to 2 thresholds to decide whether to halve, double, or keep constant K_p (Fig. 18.4.2). K_p can change between 1, 2, and 4. The option to keep the gain constant improves the performance by removing unnecessary or incorrect gain changes. The jitter tolerance of this adaptive-gain CDR is equivalent to the first-order CDR with gain of 1 at low frequency and gain of 1/4 at high frequency (Fig. 18.4.5(b)), thus breaking the link between high-frequency noise filtering and lock-time/frequency-range.

Even if these CDRs are able to operate at a given frequency offset, their jitter tolerance is degraded as the offset is increased (Fig. 18.4.5(c)). The cause of this degradation is that the wrong estimate for future phase position is being used. With a frequency offset, the future phase is not expected to be constant over time. A 2nd-order estimator that acquires the frequency offset and uses it to predict the timing of future bits corrects this error and decouples the constraints of jitter filtering and ppm tracking [3]. Previous 2nd-order CDRs used pulse/pattern generators at the input of the phase accumulator to limit the number of steps the phase DAC can make every clock cycle [2,3,4]. By allowing the integral control to update multiple phase DAC steps, this component can be removed resulting in the simpler 2nd-order CDR of Fig. 18.4.3. When updating by multiple steps, the phase control signals must be retimed close to the phase DAC and aligned to the timing sample clocks rather than those for data recovery to minimize the impact on BER. As expected, the jitter tolerance of this CDR does not degrade significantly with frequency offset (Fig. 18.4.5(d)) regardless of its bandwidth. The slight penalty seen in Fig. 18.4.5(d) is from the loop driving the error to zero only at the slow logic clock cycle such that the residual phase drift caused by the frequency offset is uncorrected for bits in between.

The estimation approach for CDRs can also be applied to systems that use a spread spectrum clock (SSC). In SSC, the nominal frequency of the TX can vary by 5000ppm in a triangular profile with a modulation frequency of 30 to 33kHz. While a 2nd-order CDR has been proposed for this system [4], the low bandwidth desired for jitter filtering compromises its ability to track the SSC. Again, these opposing constraints can be decoupled if a CDR capable of predicting the phase movement due to SSC is built, as shown in Fig. 18.4.4.

A 3rd-order estimator can predict the phase of future bits within each linear section of the triangular profile by estimating the phase, frequency, and frequency-ramp rate of the TX. However, it will perform worse than a 2nd-order when the frequency-ramp rate changes polarity. Thus, the CDR must also estimate when the frequency ramp changes sign. There are many ways to extract this information. In the proposed design, the output of the frequency accumulator (F_acc) is differentiated by subtracting each sample from the previous one. The sign of this subtraction indicates the polarity of the frequency ramp and draws out a square wave whose transitions coincide with the switching points of the SSC. Unfortunately, since the ramp rate is not large, glitches can occur in random locations. Most can be removed by running this pseudo-differentiation at a much slower sample rate such that the change in frequency will be larger between samples while others are removed by a filter with hysteresis. A digital PLL (DPLL) locked to this output is used to enable one of two frequency ramp accumulators (Rp_acc, Rn_acc). In steady state, this system operates as a 3rd-order CDR with Rp_acc half of the time and Rn_acc the other half. The use of a single ramp accumulator was investigated, but its ramp rate does not converge as the error from the positive ramp cancels out that from the negative. Saturating the ramp accumulators so that their values stay within their expected polarity helps convergence. After tape out, a more robust method of extracting the timing of the modulation is found. This method compares F_acc with its average to generate a square wave that is 90° offset from the switching points. A DPLL is locked to this signal and its output is digitally phase shifted by 90° to estimate when to switch the sign of the frequency ramp.

Figure 18.4.6 shows the timing margin of the 2nd-order and the higher-order CDR as the integral gain is varied when receiving SSC data (BER=10^{-11}). An optimum exists for the 2nd-order estimator because of the opposing bandwidth constraints from jitter filtering and tracking the SSC. The functionality of the higher-order estimator is verified by checking that its states converge to the expected values. As expected, the margin of the higher-order estimator improves at lower gain settings. Unfortunately, due to a very slow beat frequency (several millions of bits, so it was not seen in simulations) generated by the estimation of the switching points, the margin improvement is limited. Simulations show that removing this bug will result in a timing margin that increases monotonically with the reduction in integral gain giving around $0.1UI_{pp}$ overall margin improvement.

Acknowledgment:
Authors thank MARCO for funding and D. Liu and D. Draper for assistance.

References:
[1] S. Sidiropoulos, et al., "A Semidigital Dual Delay-Locked Loop," *IEEE J. Solid-State Cicuits*, Nov., 1997.
[2] H. Lee, et al., "Burst Mode Packet Receiver Using a Second Order DLL," *Dig. Symp. VLSI Circuits*, vol. 32, pp. 1683-1692, June, 2004.
[3] H. Ng, et al., "A Second-Order Semidigital Clock Recovery Circuit Based on Injection Locking," IEEE *J. Solid-State Circuits*, vol. 38, pp. 2101-2110, Dec., 2003.
[4] M. Aoyama, et al., "3Gbps, 5000ppm Spread Spectrum SerDes PHY with Frequency Tracking Phase Interpolator for Serial ATA," *Dig. Symp. VLSI Circuits*, June, 2003.

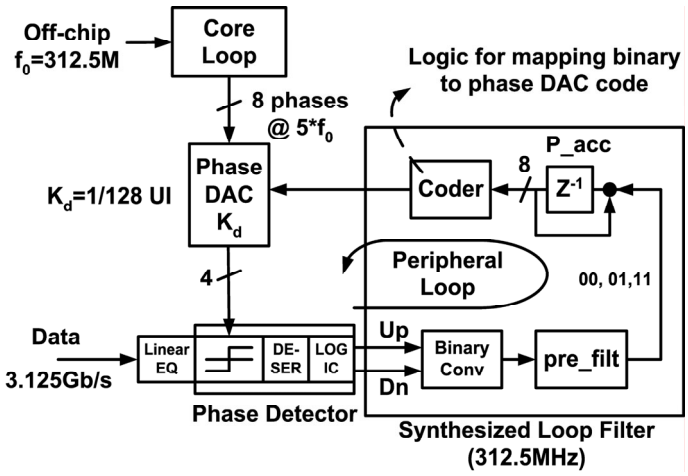

Figure 18.4.1: First-order semi-digital dual-loop CDR.

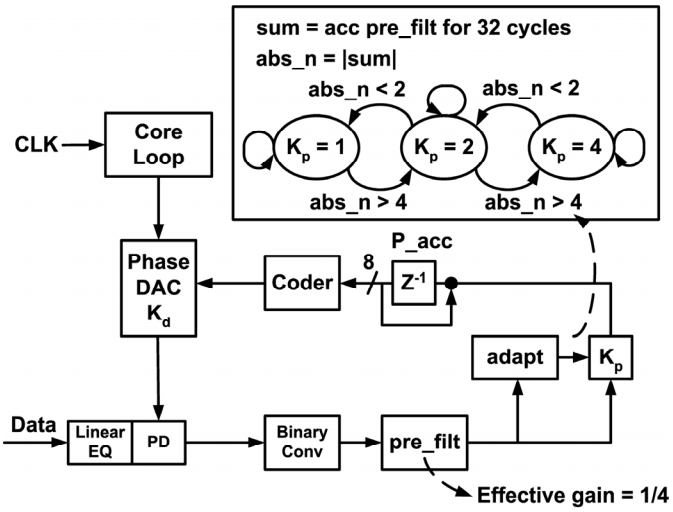

Figure 18.4.2: First-order CDR with adaptive gain.

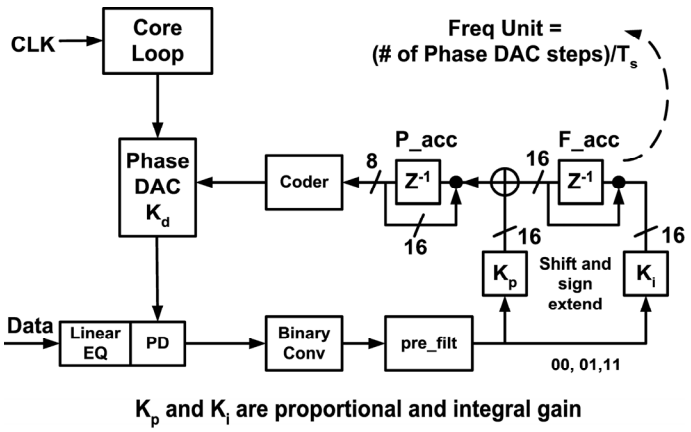

Figure 18.4.3: Second-order CDR capable of tracking > 5000ppm.

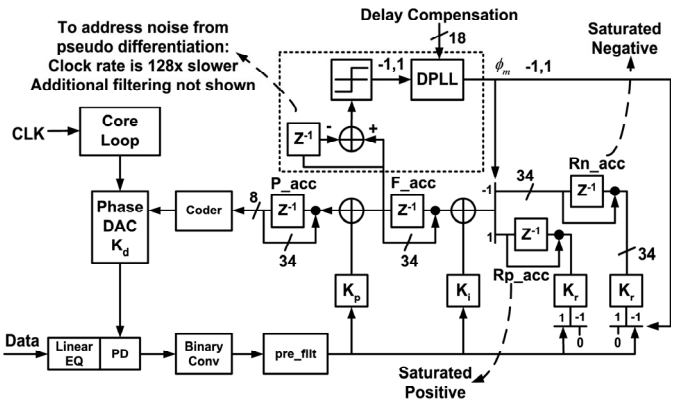

Figure 18.4.4: Higher-order CDR for recovering SSC data.

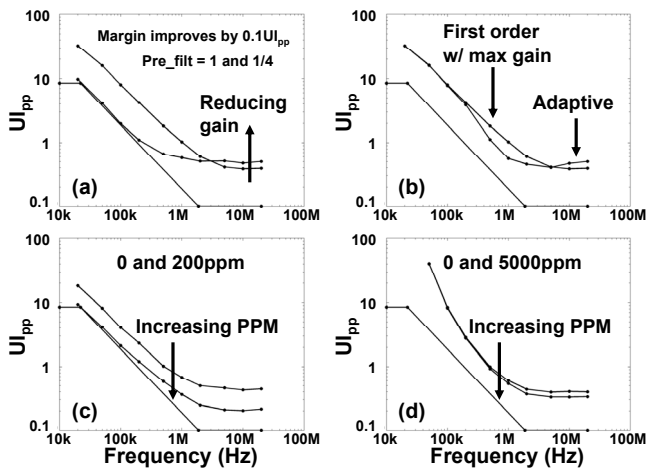

Figure 18.4.5: Jitter tolerance measured according to XAUI specs. (a) first-order CDR vs. bandwidth, (b) first order with adaptive gain, (c) first order vs. ppm, and (d) second order vs. ppm.

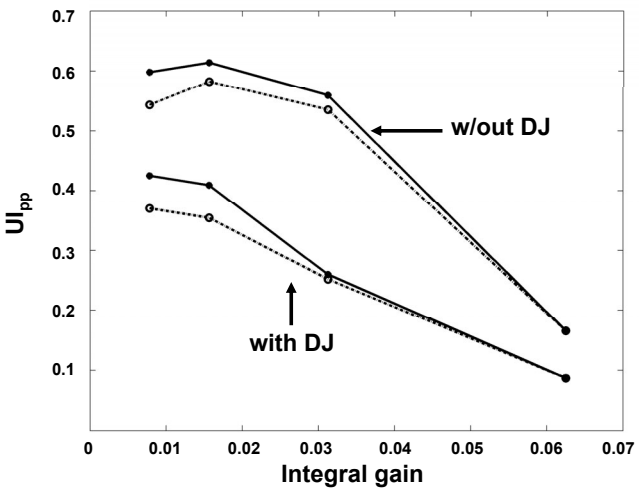

Figure 18.4.6: Timing margin (UI_{pp}) as integral gain is varied with SSC data (BER=10^{-11}) for the second-order (dotted, °) and higher-order (solid,*) estimators.